Addressing DIO Suppression Attack in RPL based IoT Networks

Rajat Kumar, Jyoti Grover, Girish Sharma, and Abhishek Verma

Abstract The Internet of Things (IoT) has brought a revolution in technology in the last decade. IoT is susceptible to numerous internal routing attacks because of the characteristics of the sensors used in IoT networks and the insecure nature of the Internet. The majority of the IoT ecosystem's problems come during the routing phase. While routing, the attacking node causes a number of challenges with the packet transmission mechanism. Routing Protocol for Low-Power and Lossy Networks (RPL) is susceptible to numerous types of attacks. The effects could be disruptive to network performance and resource availability. In this paper, we investigate the impact of a novel attack known as the DIO suppression attack and propose a mitigation mechanism for this attack on RPL-based network. This attack disrupts the topology of a network, and as a result, certain number of nodes are disconnected. Attacker nodes exploit the trickle algorithm to execute this attack. The impact of DIO suppression attack in different topologies and scenarios is studied in this research. We have also proposed a lightweight mitigation technique to defend the networks from this attack. This technique leverages the trickling timer's DIO Redundancy Constant *k* for each node to identify the attacking node in the network.

Keywords IoT · RPL · DIO suppression attack · Security · Routing attack

R. Kumar · J. Grover (✉) · G. Sharma
Malaviya National Institute of Technology Jaipur, Jaipur 302017, India
e-mail: jgrover.cse@mnit.ac.in

R. Kumar
e-mail: 2020rcp9012@mnit.ac.in

G. Sharma
Manipal University Jaipur, Jaipur 303007, India

A. Verma
Babasaheb Bhimrao Ambedkar University, Lucknow 226025, Uttar Pradesh, India
e-mail: abhiverma@iiitdmj.ac.in

S. J. Patel et al. (eds.), *Information Security, Privacy and Digital Forensics*,
Lecture Notes in Electrical Engineering 1075,
https://doi.org/10.1007/978-981-99-5091-1_8

1 Introduction

The term Internet of Things (IoT) refers to a network of interconnected devices that are built with sensors, software, and other technologies to transmit and receive data to and from other devices. IoT is used in a variety of industries, each with its own set of security concerns, including health care, smart homes, and autonomous cars. IoT devices are susceptible to various security attacks because of their resource restrictions if they are connected to one another via lossy communication networks.

The majority of Internet of Things applications are made possible by the widespread use of IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN) [15], a form of Low-Power and Lossy Networks (LLNs). RPL was created by the IETF [12] Routing Over Low-power and Lossy Networks working group (ROLL) that provides routing functionality in LLN. Traditional routing protocols are inappropriate for LLN due to its features [9]. A network may be vulnerable to different routing problems from both internal and external attackers due to insufficient security. RPL is vulnerable to various security attacks that can affect the security and privacy of its users because of its self-organization, self-healing, openness, and resource-constrained nature. Majority of security solutions concentrate on applying cryptographic techniques to secure the RPL control messages. However, if the encryption keys have already been compromised, cryptographic methods cannot defend the network from inside attackers [5]. By utilizing the hacked nodes, internal attackers have the ability to forcefully reduce the network performance and control communication.

In this paper, the DIO suppression attack and its effects on RPL-based networks are examined, and a mitigating method is proposed. This attack disrupts the topology of the network by exploiting trickling algorithm. Therefore, certain number of sensor nodes get disconnected in the network. DIO suppression attacks have the potential to drastically reduce the Average End-to-End Delay (AE2ED), Average Power Consumption, and Packet Delivery Ratio (PDR) of RPL-based networks.

In a DIO Suppression Attack, a malicious node broadcasts DIO messages to legitimate nodes. If the attacker node sends same DIO packet consistently [7], legitimate receiver nodes start suppressing their own DIO transmission which is governed by trickle algorithm [4]. Because DIO packets are used to identify neighbors and network topology, their suppression may result in network partition and some routes may remain undiscovered. The contributions of this paper are listed below:

- On RPL-based IoT networks, a comprehensive analysis on the impact of the DIO suppression attack in various topologies and circumstances is conducted.
- A lightweight mitigation technique to address DIO suppression attack is presented. This method leverages the trickling timer's DIO redundancy constant k for each node to identify the attacking node in the network.

The remaining paper is organized as follows. Section 2 presents the working of RPL protocol. In Sect. 3, we describe related work. The DIO suppression attack is presented in Sect. 4. A detailed discussion of the experimental evaluation of the DIO suppression attack is presented in Sect. 6. A lightweight solution to address DIO suppression attack is discussed in Sect. 7. The paper is concluded in Sect. 8.

2 Background

This section presents the overview of RPL protocol and DODAG construction.

2.1 Introduction to RPL Protocol

RPL protocol can provide routing capability in LLNs, where devices are severely resource constrained. Network devices are arranged into Directed Acyclic Graphs (DAGs) by distance-vector routing technique. A network that has no round-trip routes between any two nodes is referred to as a DAG. Then, traffic is directed toward one or even more DODAG root. The DODAGs, which are directed acyclic networks only with single root node and sink all data, are contained within the DAG. One or more DODAGs may be present in each of the numerous RPL instances that coexist inside a DAG, enabling several applications to run concurrently and independently over the network.

Internet control management protocol version 6 (ICMPv6) is the foundation for RPL control messages [2]. Following control messages are used by RPL in DODAG construction, DODAG Information Object (DIO, DODAG Information Solicitation (DIS), Destination Advertisement Object (DAO), Destination Advertisement Object Acknowledgment (DAO-ACK), and Consistency Check (CC).

2.2 DODAG Construction and Working

Exchanges of DIO messages are used to construct DODAGs, and this process always begins at the root node. The majority of the DIO base fields, including DODAG version, DODAG ID, RPL instance ID, and RPL mode of operation, is set by the root node. When a DIO message is received, each node determines its rank with the help of specified Objective Function. Figure 1 represents different steps of DODAG construction.

When parents are selected based on rank, routing loops are avoided. DIO messages are always exchanged frequently to maintain the routing topology, and nodes may choose to discard a new DIO message if it does not cause any changes in the current DODAG scenario at the receiving node (such as a change in the DODAG version number) or a change in the node's preferred parent. RPL employs the Trickle algorithm to limit the quantity of DIO messages in order to preserve the limited resources of nodes. Each node maintains a DIO counter and a timer with a threshold value. When the trickling game concludes or when a DIO message that updates the RPL configuration is received, a DIO message is dispatched.

The DIO packet count will now be raised each time a DIO packet is transmitted and ignored. The count and trickle timer are both reset and the trickling time is

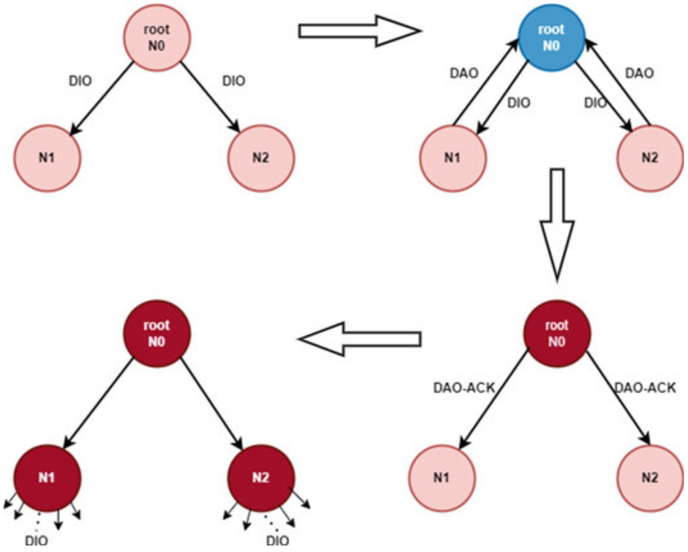


Fig. 1 DODAG construction

doubled if the counter hits the threshold number. Additionally, when a modification is made as a result of a DIO message received, the DIO count and trickle timer will reset to their initial values. Whenever the network is stable, this approach enables fewer DIO message broadcasts and enables rapid topology updates when there are changes.

3 Related Work

The authors [7] suggested two mitigation mechanisms for the DIO suppression attack. The first is inserting the message integrity code into the messages specified by the RPL specification. The second is to implement MAC-layer encryption. The latter approach uses more computing power and increases network traffic overhead.

Yavuz et al. [16] presented a deep learning-based IDS against version number and hello flood attacks on RPL-based IoT networks. The authors suggested five hidden layers in a neural network. They used the Cooja simulator to simulate networks with 10 to 1,000 nodes. Their experimental findings for version number attack and hello flood attack revealed the precision and recall of 94% and 97%, respectively. The published study did not, however, include the false-positive rates.

Mayzaud et al. [6] described the hybrid placement IDS for version attack. Numerous "monitoring sensors" are strategically placed throughout the network to monitor DIO messages. The IDS underwent assessment by the writers. Their trial's findings showed high detection rates. Additionally, it was shown that false-positive detection might be decreased.

Sedjelmaci et al. [10] describe the distributed placement IDS. Their theory is that signature-detection can detect frequent attacks while anomaly-detection is only done when malicious traffic is identified. The authors combined their methodology with a scheme to limit the number of false positives. The IDS was subject to a sinkhole attack test. The evaluation produced comparable results to SVELTE while consuming less energy.

The Parent Failover and Rank Authentication techniques, covered in [14], protect against the Sinkhole attack. The first method uses a one-way hash that is created and added to DIO messages to enable genuine nodes to determine whether another node on the route to the sink is inadvertently reporting a rank. In the latter, a sink node tells a child node that it is not delivering enough traffic (based on a predetermined baseline).

The authors [1] examined several variables, along with the periodicity of DIO packets, packet delivery ratio, and packet loss, to examine the impact of black-hole attacks. Additionally, they proposed a protection system based on a per-node scheme based on the forwarding habits of network neighbors.

In [3], the authors offered a variety of wormhole attack detection methods. Giving the nodes and, by extension, the neighborhood geographic information is one strategy. Another choice is to use various link layer secret keys for every network segment, which prevents communication between two nodes in different parts. It is more challenging to use a Merkel tree authentication schema to build the topology.

In [13], it has been proposed to counteract the selective forwarding attack by establishing alternate paths inside the RPL topologies that are dynamically selected by nodes.

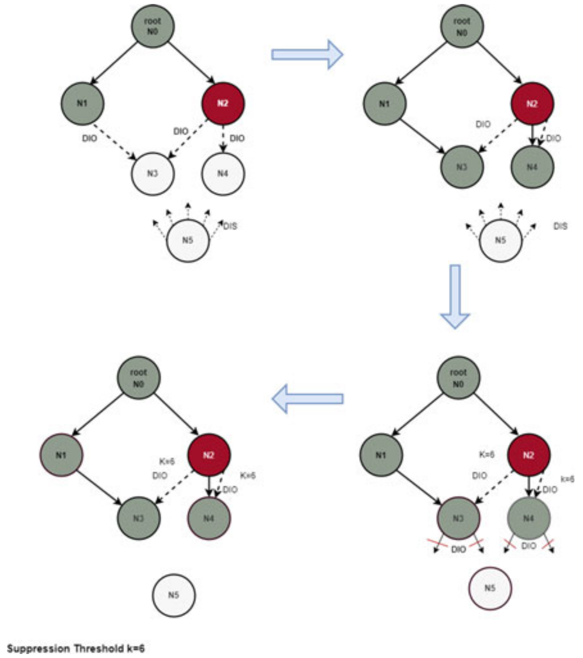
Reference [8] claims that almost no special defence against the Hello Flood attack has been developed. The simulation shows how the RPL Global and Local repairing mechanisms will quickly deal with the attacker.

4 DIO Suppression Attack Overview

A malicious node broadcasts DIO messages to legitimate nodes in a DIO suppression attack. If the attacker node sends the same DIO packet repeatedly, the recipient nodes consider it consistent[7]. If they receive consistent DIOs, nodes will suppress their own DIO transmission, which is governed by the trickle algorithm[4]. Because DIO packets are used to identify neighbors and network architecture, their suppression may result in some nodes remaining hidden and some routes remaining undiscovered. Attacks on DIO suppression harm the performance of IoT network protocols like the RPL protocol.

The transmitter in RPL broadcasts the DIO while DODAG is being created. Once the receiver has received the DIO from the transmitter, it adjusts its sibling list and parent list rank and transmits a DAO packet with route information. A malicious node will repeatedly send DIO messages to legitimate nodes after receiving it. Honest nodes will stop transmitting DIOs when they get a DIO packet from a malicious node.

Fig. 2 Working of DIO suppression attack during DODAG formation



As a result of continuous suppression, some nodes might continue to be hidden, and some routes might continue to be undiscovered.

In Fig. 2, N0(root), N1, N2, N3, N4, and N5 nodes are available to construct a DODAG to transmit the data between the nodes.

N0 initiates the construction of the DODAG by broadcasting the DIO message to the nearest nodes. N1 and N2 receive the DIO messages from N0. N1 and N2 acknowledge the DIO message with the DAO control messages, and N0 sends back another DAO-ACK message as an acknowledgement. Now, N1 and N2 are connected to node N0. N1 and N2 transmit the DIO messages to join the other nodes in the network. N2 is a malicious node in this network. N2 then sends the DIO message to the nodes that want to join the network. But this attacking node is programmed to send the same DIO message every time. N2 sends a DIO message to N3 and N4. We have set the DIO redundancy constant (threshold) to 6. So N3 and N4 will get the same DIO message. If N3 and N4 receive the six consistent DIO messages then these nodes will not transmit the DIO message in future.

5 Experimental Setup

This section discusses the impact analysis of a DIO suppression attack based on simulation. Using the NetSim simulator, a number of sets of experiments were conducted to examine the impact of a DIO suppression attack on an RPL-based network [11],

Table 1 Parameters for simulation model

Parameter	Value
Simulator	NetSim
Topology	Grid, Random
Number of nodes	5, 10, 15, 20, 25, 30
Number of nodes in grid	16
Number of malicious nodes	10%, 20%, 30% of legitimate nodes
Routing protocol	RPL protocol
Area	500 m * 500 m
Simulation time	100 s
Transmission range	50 m
Interference range	100 m
Data packet size	127 Bytes
Mobility	Random mobility

which is the most reliable and widely used network simulator. Table 1 presents simulation parameters considered in various experiments.

Two network topologies are used to simulate this attack: (1) grid topology and (2) random topology. For grid topology, we took 16 nodes and compared them with the 16-node random topology. In other scenarios, we took 5, 10, 15, 20, 25, 30 nodes and varying numbers of malicious nodes, i.e., 10%, 20%, 30% malicious nodes. All these simulations are done for static and mobile nodes and compared with each other, which is discussed in the Results and Analysis section of this paper.

6 Results and Analysis

The findings and analysis of the simulation is presented in this section. The attack's impact is evaluated using three parameters: throughput, average battery consumption, and delay.

Throughput—The amount of data moved successfully from one place to another in a given time period, and it is measured in kilo bits per second (kbps).

Delay—It represents the typical time required for all packets to travel from the source application to the target application layer.

Average Battery Consumption—It displays the average battery consumption over the whole network of connected nodes.

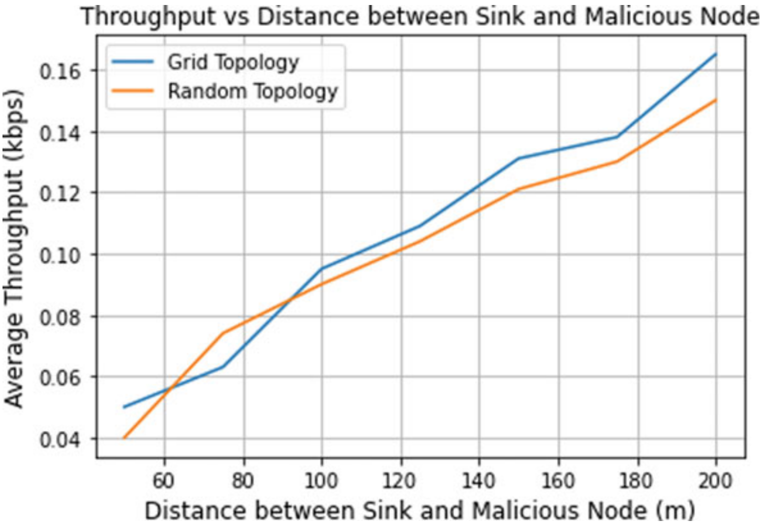


Fig. 3 Comparison of throughput and distance between sink node and malicious node in 16-node grid and random topology

6.1 Impact of Distance Between Malicious and Sink Node on Throughput

In the DIO suppression attack, some nodes remain disconnected because of the suppression of DIO control messages, which are responsible for the construction of DODAG in routing. Figure 3 shows that throughput decreases if the distance between the sink and malicious node decreases, i.e., if the attacker node is near the sink node, then this attack is more fatal.

Throughput drops in the random topology. Random topology increases the probability of disconnection caused by an attacker node, which causes more packet losses in the network and a reduction in performance. Figure 4 shows that throughput is decreased exponentially if nodes are mobile.

6.2 Impact of Varied Malicious Nodes in Different Topologies on Throughput

Our analysis demonstrate that the throughput decreases as the malicious nodes in the topology increase. Figure 5a illustrates the effects of malicious nodes in a static scenario with percentages of 10%, 20%, and 30% in networks of 5, 10, 15, 20, 25, and 30 nodes.

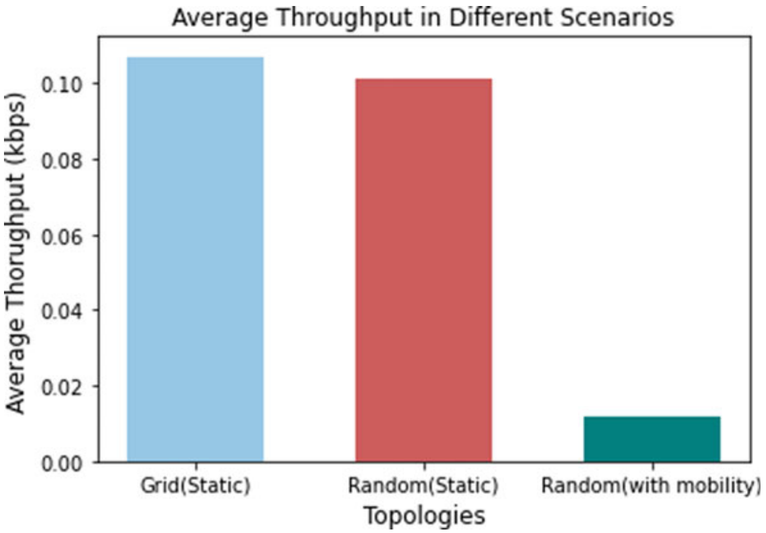
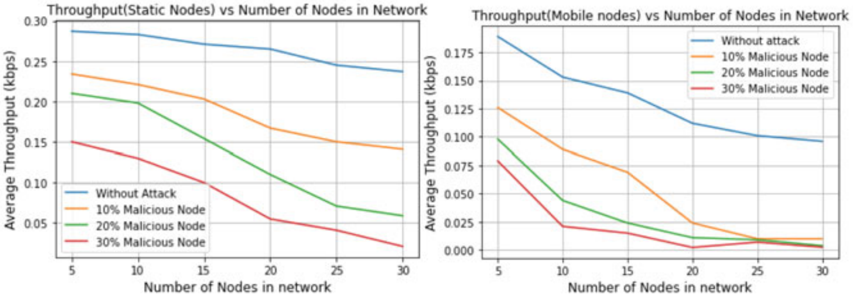


Fig. 4 Average throughput in different scenarios and topology of 16 nodes



(a) Throughput in Static Scenario vs Number of Nodes in Network

(b) Throughput in mobile scenario vs Number of Nodes in Network

Fig. 5 **a** Throughput in static scenario versus number of nodes in network, **b** Throughput in mobile scenario versus number of nodes in network

If all nodes are mobile, the DIO suppression attack would become more severe. If all nodes were mobile with an increased number of malicious nodes, throughput would further be significantly reduced as can be seen in Fig. 5b.

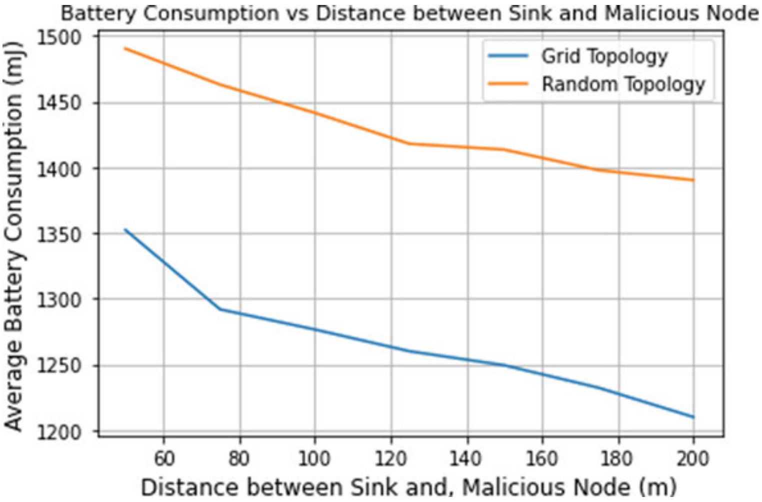


Fig. 6 Battery consumption versus distance between the malicious node and sink node graph

6.3 Impact of Distance Between Malicious and Sink Node on Battery Consumption

This section analyzes the battery usage of different topologies and scenarios. As malicious node moves away from the sink node, less average battery power is used as can be seen in Fig. 6. The average battery consumption is higher if the malicious node is close to the sink node. Figure 7 shows the battery consumption between grid and random topology. The average battery consumption in random topology is greater than in grid topology.

If nodes are mobile, then the battery consumption is highest because DODAG construction is more frequent in a mobile scenario that needs more processing power, so battery consumption is increased.

6.4 Impact of Varied Malicious Nodes in Different Topologies on Battery Consumption

Battery consumption increases if the number of total and malicious nodes increases in the network. From Fig. 8a, we can analyze the attack's impact on battery consumption. The battery consumption will increase if the number of nodes in the network increases. If we change the malicious nodes from 10% to 20%, battery consumption increases.

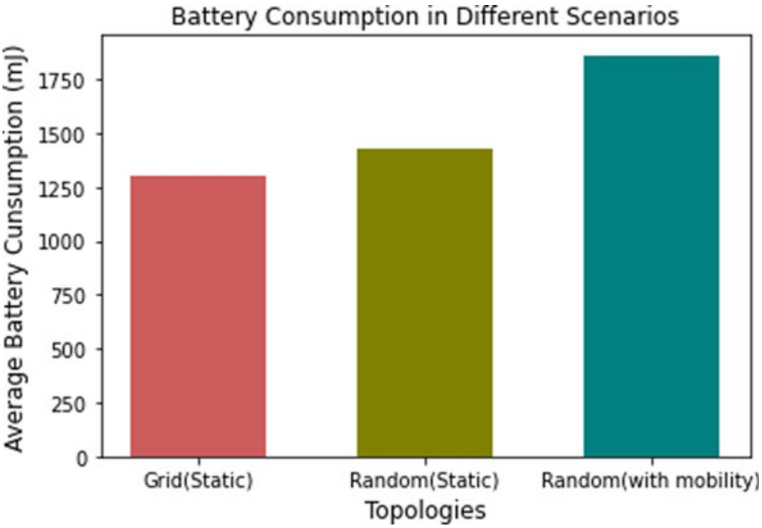
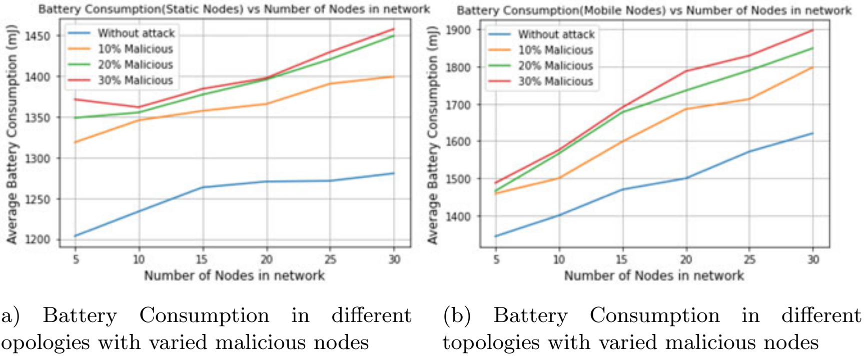


Fig. 7 Average battery consumption in different scenarios and topologies



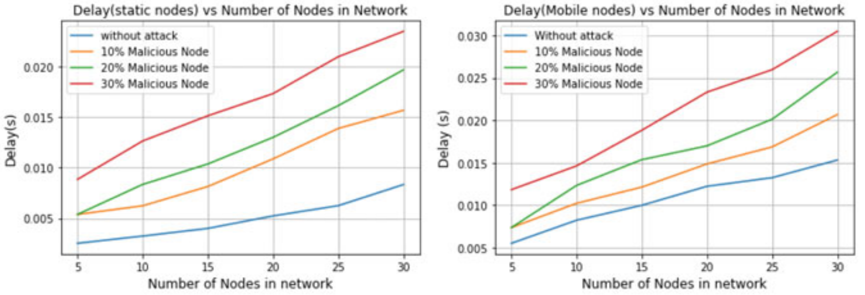
(a) Battery Consumption in different topologies with varied malicious nodes

(b) Battery Consumption in different topologies with varied malicious nodes

Fig. 8 **a**, **b** Battery consumption in different topologies with varied malicious nodes

6.5 Impact of DIO Suppression Attack on Delay in RPL-Based Networks

In this section, we will compare the delay for different scenarios and topologies in the network. Delay will increase if the number of attacking nodes increases in the network. DIO suppression attack disconnects the nodes which result in the increase of delay. Figure 9a shows that in the static scenario for 10%, 20%, 30% malicious nodes in the network of 5, 10, 15, 20, 25, 30 nodes, the delay is increasing significantly. Increasing delays affect the communication between the nodes during the route, which gives less throughput.



(a) Delay in different topology with varied number of Malicious nodes

(b) Delay in different number of nodes and different number of malicious nodes

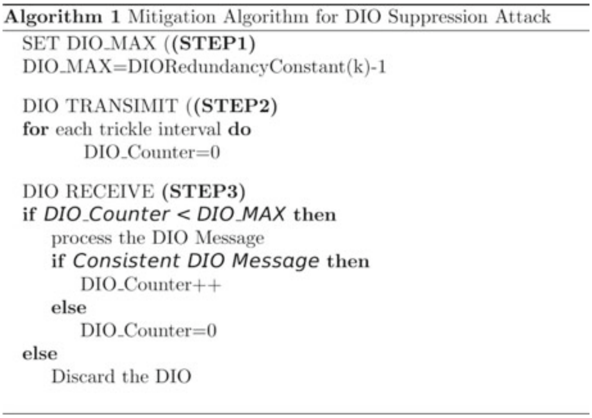
Fig. 9 **a** Delay in different topologies with varied number of malicious nodes, **b** Delay in different number of nodes and different number of malicious nodes

If nodes are mobile, then this attack shows more fatal results. If nodes are moving, then a malicious node can get in touch with the greater number of nodes in the network. It affects the topology of the network, and more nodes remain disconnected, which results in higher delays in the network. As we can see in Fig. 9b for the mobile scenario, in the network of 5, 10, 15, 20, and 30 nodes with 10%, 20%, 30% of malicious nodes, the delay increases significantly. The delay also increases if the number of nodes increases.

7 Mitigation Mechanism for DIO Suppression Attack

As we have seen in the result and analysis section, this attack is becoming more fatal if the number of malicious nodes increases or if malicious nodes get closer to the sink node. If nodes are mobile, then the negative impact of this attack will increase exponentially. In this section, we will propose a mitigation mechanism for this attack. This mitigation mechanism is a frequency-based solution to mitigate and detect this attack.

This solution works on every node during the transmission of the data packets. In the first step of this mechanism, we set a DIO_MAX, which is less than the threshold value to suppress the DIO message for any node. The second step is started when a node transmits a DIO message. There is a trickle timer which is working for every node. The trickle algorithm divides the time into variable trickle intervals. If a node receives consistent messages equal to the threshold value, then the trickle algorithm suppresses the transmission of the DIO messages from that node.



Algorithm 1 Mitigation Algorithm for DIO Suppression Attack

```
SET DIO_MAX ((STEP1)
DIO_MAX=DIORedundancyConstant(k)-1

DIO TRANSIMIT ((STEP2)
for each trickle interval do
        DIO_Counter=0

DIO RECEIVE (STEP3)
if DIO_Counter < DIO_MAX then
    process the DIO Message
    if Consistent DIO Message then
        DIO_Counter++
    else
        DIO_Counter=0
else
    Discard the DIO
```

In the second step, for each trickle timer, we will set our DIO_Counter for every trickle interval; for every trickle interval, it will count the number of consistent messages. In the third and essential step, it will check if the DIO_Counter is less than the DIO_MAX then it will process the DIO packet. After processing, if that packet is consistent, it will increment the DIO_Counter; otherwise, it will make DIO_Counter zero because of the received inconsistent packet. If DIO_Counter becomes equal to the DIO_MAX, it will discard the next receiving packet because the next packet may be a consistent message which can suppress the DIO of the child node. Figure 10 shows the working of the mitigation mechanism for the proposed algorithm for RPL-based IoT networks.

8 Conclusion and Future Work

In this paper, we have presented the analysis of DIO suppression attack in different scenarios of a RPL-based IOT network. This attack is more severe if the attacker is present near the sink node. Also, if the number of malicious nodes increases in the network, this attack becomes more fatal. In this attack, the victim node suppresses its DIO messages because it gets consistent messages from the malicious node that equals the threshold value. We have also analyzed this attack on mobile networks, and the results are more fatal on mobile networks. We have also proposed a countermeasure for this attack which may reduce the risk of the legitimate node becoming a victim node of this attack.

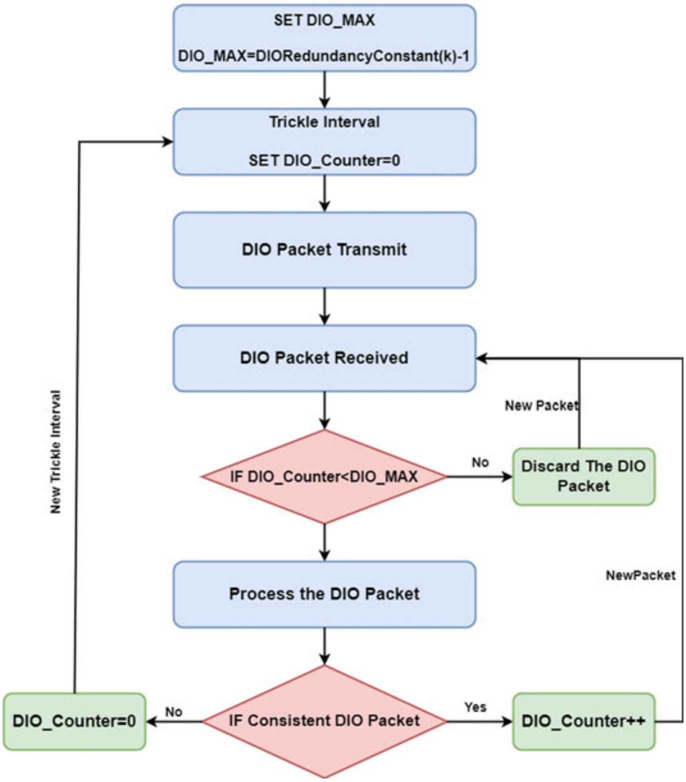


Fig. 10 Working of mitigation mechanism for DIO suppression attack

References

1. Airehrour D, Gutierrez J, Ray SK (2016) Securing RPL routing protocol from blackhole attacks using a trust-based mechanism. In: 2016 26th International telecommunication networks and applications conference (ITNAC). IEEE, pp 115–120
2. Cao Y, Muqing W (2018) A novel RPL algorithm based on chaotic genetic algorithm. Sensors 18 (10):3647
3. Khan FI, Shon T, Lee T, Kim K (2013) Wormhole attack prevention mechanism for RPL based LLN network. In: 2013 Fifth international conference on ubiquitous and future networks (ICUFN). IEEE, pp 149–154
4. Levis P, Clausen TH, Gnawali O, Hui J, Ko J (2011) The trickle algorithm. RFC 6206
5. Miloslavskaya N, Tolstoy A (2019) Internet of Things: information security challenges and solutions. Clust Comput 22(1):103–119
6. Mitra D, Gupta S (2021) Data security in IoT using trust management technique. In: 2021 2nd International conference on computational methods in science & technology (ICCMST) (Los Alamitos, CA, USA, Dec 2021). IEEE Computer Society, pp 14–19
7. Perazzo P, Vallati C, Anastasi G, Dini G (2017) DIO suppression attack against routing in the internet of things. IEEE Commun Lett 21(11):2524–2527
8. Pongle P, Chavan G (2015) A survey: attacks on RPL and flowpan in IoT. In: 2015 International conference on pervasive computing (ICPC). IEEE, pp 1–6

9. Raoof AM (2021) Secure routing and forwarding in RPL-based internet of things: challenges and solutions. PhD thesis, Carleton University
10. Sedjelmaci H, Senouci SM, Taleb T (2017) An accurate security game for low-resource IoT devices. IEEE Trans Veh Technol 66(10):9381–9393
11. Tetcos. Tetcos: Netsim—network simulation software, India
12. Vasseur J (2014) Terms used in routing for low-power and lossy networks. RFC 7102
13. Wallgren L, Raza S, Voigt T (2013) Routing attacks and countermeasures in the RPL-based internet of things. Int J Distrib Sens Netw 9(8):794326
14. Weekly K, Pister K (2012) Evaluating sinkhole defense techniques in RPL networks. In: 2012 20th IEEE International conference on network protocols (ICNP). IEEE, pp 1–6
15. Winter T, Thubert P, Brandt A, Hui JW, Kelsey R, Levis P, Pister K, Struik R, Vasseur JP, Alexander RK et al (2012) RPL: IPv6 routing protocol for low-power and lossy networks. RFC 6550:1–157
16. Yavuz FY, Ünal D, Gül E (2018) Deep learning for detection of routing attacks in the internet of things. Int J Comput Intell Syst 12:39–58